DOCUMENT RESUME

ED 427 683 IR 019 234

AUTHOR Bollin, Andreas
TITLE Improving Lectures and Practical Classes in Using an Automatically Feedback System.
PUB DATE 1998-11-00
NOTE 7p.; In: WebNet 98 World Conference of the WWW, Internet, and Intranet Proceedings (3rd, Orlando, FL, November 7-12, 1998); see IR 019 231.
PUB TYPE Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Automation; Computer System Design; *Computer Uses in Education; *Course Evaluation; Data Analysis; Evaluation Methods; Feedback; Foreign Countries; Higher Education; Instructional Effectiveness; *Instructional Improvement; Instructional Materials; Lecture Method; Questionnaires; *Student Evaluation of Teacher Performance; *Systems Development; World Wide Web
IDENTIFIERS Austria; *Automated Feedback; Client Server Computing Systems; Java Programming Language; Object Oriented Programming

ABSTRACT

This paper presents a World Wide Web-based electronic feedback system for use in lectures and practical classes. The system is based on Java and provides a configurable feedback form, a managing tool for administrators, and a statistics viewer for presenting the generated statistical data in various ways. In addition, it generates a statistical history, which is an important instrument in comparing lectures and in tracing effects of improvements or alterations of the lecture. This paper shows how anonymity can be guaranteed to the user and outlines the resulting problems. It concludes with possible improvements to prepare the system for more general use. (Author/AEF)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 427 683

Improving Lectures and Practical Classes in using an Automatically Feedback System

Bollin Andreas, bollin@ist.tu-graz.ac.at
IICM Software Technology
Technical University Graz, Austria

Abstract: This paper presents an electronically feedback system for use in lectures and practical classes. The system itself is based on Java and provides a configurable feedback form, a managing tool for administrators and a statistic viewer for presenting the generated statistical data in various ways. In addition it generates a statistical history, which is an important instrument in comparing lectures and in tracing effects of improvements (or alterations) of the lecture. The paper shows how anonymity can be guarantee to the user and outlines the resulting problems. It concludes with possible improvements to prepare the system for a more general use.

1 Introduction

Because of today's big amount of students, teaching and working is, for both sides (tutors and students), hard effort. Materials are often that extensive, that lecturers can only teach parts out of it and only can help students in understanding the whole material. The more it is important to react to students' wishes and to understand their problems as early as possible. A crucial part of a lecturer's work is keeping contact to the students (during the lectures, office hours, email, news groups). In addition it is usual to use feedback forms to produce statistics over the lecture. On many universities (as in private companies) feedback is an important tool for comparing lectures and lecturers (or institutions). Ignoring this way of assessment is therefore not only a way of ignorance, it is also stupidity.

Usually at the end of semester feedback sheets are distributed and then again collected. With a one or two days effort (depending on the amount of students) all the forms are analysed and statistical data is generated. In most cases the data is not electronically stored and a comparison between different lectures is not possible. This approach has the following main disadvantages:

- too much time is needed for analysing the data,
- students are (generally) not informed about the results of the feedback,
- possible problems within the lecture are found *too* late.

It turns out, that only a few points should be altered which then is leading to an improvement of the lecture (see also [Lesgold 98]):

- the feedback should be analysed automatically,
- the data should be available electronically (for later evaluation),
- statistical history should be available,
- the data should be able to be accessed or entered from (nearly) everywhere,
- students should stay anonymous whenever possible (especially when entering feedback data),
- students should see the results of the feedback,
- feedback should be available as early as possible (during the lecture).

IR019234

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

BEST COPY AVAILABLE

PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
G.H. Marks
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC).

Finding solutions for that demands will lead to more motivated students, better lectures and less work for the lecturers. The system itself has to be that interesting, that both, students and lecturers, like to use it.
Statistical history is a powerful tool in tracing the development of lectures. As „Illustration is the basis of all cognition's" (Pestalozzi) graphical presentation of statistical data helps in understanding the results of the assessment and illustrates the (hopeful) improvement of the lecture.

This paper presents an approach of an electronically feedback system using the World Wide Web (WWW for short) and Java that is trying to fulfil all the demands listed above. The whole work is based on a project [Bollin, Luidolt 95] in 1995, where a feedback system was analysed using an object oriented approach (the Jacobson Method). With Java 1.0 (in 1996/97) it became possible to implement the system over the net and the first try run started in the winter term 1997/98.

2 Strategy for Improvement

At the beginning of the project our students had to fill out the feedback form in one of the last lectures of a term. The data was analysed and the results presented (and discussed) in the last lecture. Strongly motivated from the results of the discussion but stressed from the work of analysing hundreds of feedback forms our institute decided to provide a further service to the students: an *electronically* feedback form. In 1995 (based on the theoretical study of such a distributed system [Bollin, Luidolt 95]) we implemented parts of it in HTML and used CGI scripts to store the data. This system did not implement the system, as it was designed in 1995. There where two main disadvantages (leading to less motivation to use it):

- no feedback generated for the students,
- statistical data was not generated automatically.

As it turned out, the system was only used by a couple of students. Most of our students did not want to spent their time in filling out forms and saw no profit in using an electronically system. For short: it was not interesting to them. As our institute began to improve its services on the net and started to use an intelligent working environment [Bollin 97], the students became more motivated.

Motivation is the key factor for both, students and tutors. As common Internet browsers became able to run Java applets, we decided to implement the whole system using Java. The system then provided not only a feedback form, but also a viewer for students to display the statistical results of the feedback. With looking at the statistical history of a lecture every student is now able to trace the public mood in a lecture - which indeed is leading to more motivation. Using this approach we were able to increase the numbers of students using the system dramatically [Tab. 1].

	1995	1996	1997	1998
Feedback Sheets	130	119	112	108
Electronic Feedback	-	8	26	120
Students total	220	194	186	184

Table 1: Filled out feedback sheets and electronic feedback forms in the lecture „Informatik I" from 1995 to 1998.

As you can see in table 1 (which is including the trial run of our system in 1997/98) the new system provided much more motivation for students than the old version (with simple scripts in the background) or the paper version. In addition to the fact that analysing the data needed no time, students were highly satisfied with the possibility in looking at the results and used the system more often.

Another problem at the beginning was the scepticism in concern with anonymity. Filling out a feedback paper during the lecture (among hundreds of other students), looked much safer for most of the students. They seemed to be afraid that their opinion might influence their marks. Therefor we decided not to implement identification at the moment.

Every user is allowed to fill out the electronic feedback form, it is also possible to fill out the form more than only once. At the first glance this seems to be a problem, because one student could influence the statistics more than it is wished to, but on the other hand this kind of influence is, to a great extend, also possible with the paper form of the feedback. The problem could only be overcome with partial identification, which will be part of one of our next releases.

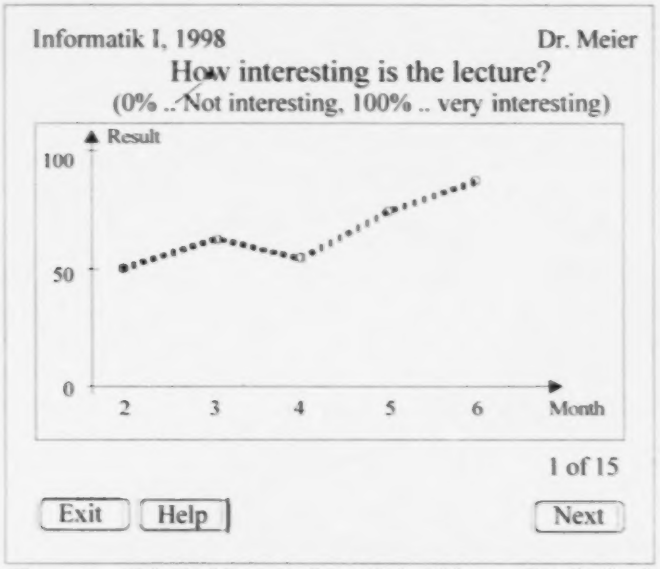

Figure 1: Principal layout of a statistical history. The feedback started in February - the history is displayed till the end of June

As mentioned above statistical history is a main factor for motivation. It is based on the fact, that feedback forms can (and should) be filled out throughout the whole lecture. The statistical history is then generated in the following manner:

1. the deadline is always the end of a month. The monthly summarising statistical data cannot be displayed for the running month,
2. the statistical data collected during the month is analysed and stored separately. This guarantees fast loading times and avoids time consuming computations,
3. the results of all months are summarised and can be displayed graphically.

Following this strategy, every month the data is collected and analysed. A lecture that starts in February and ends in June (five months) leads to six sets of statistical data - the average statistics of the whole lecture (which can be displayed at any time) and the statistics over every month (which can be displayed after the end of the first month). Displaying the results of a question (for example: „How interesting is the lecture?") leads to a good representation of the mood during the lecture (see [Fig. 1]). This graphical interpretation of the mood during the lecture could be a good basis for tutors improving their lectures.

3 Design and Implementation of the Client

The latest version of our system is now fully based on the object oriented design and was implemented in Java JDK 1.0.2. We decided not to use JDK 1.1 to be sure, that our applets are also running on older versions of Internet browsers (like Netscape). The system is designed to be used across the Internet and therefor all class files are stored in a JAR (Java Archive) file, whose size is only 76 Kbytes. The JAR file together with a starting page now residues on our Hyper Wave Server [Kappe 93] and is accessible through the Internet (http://www.ist.tu-graz.ac.at/Feedback/Feedback.html).

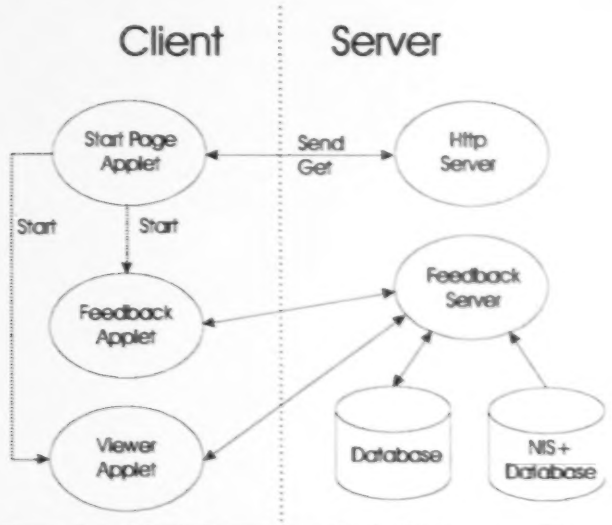

Figure 2: Principal structure of the feedback system. The feedback server has to run on the same machine as the http server, applets are started on demand.

At the moment the feedback system provides following functionality:

- feedback form for anonymous users. It provides an electronic implementation of a typical feedback sheet. The user is able to select a lecture and to fill out all or only some of the questions in the form. *No identification* is necessary to fill out the feedback form. At the moment there are two types of questions possible. Questions with radio buttons (one out of five: 0%, 25%, 50%, 75% and 100%) as a rating and questions with a textual entry field for answers or comments.

- viewer modes for anonymous and identified users. The viewer modes for anonymous and authorised users differ a little bit. Without identification every user is able to look at the statistical results. Anonymous users can get numerical and graphical feedback of a lecture and are able to trace its statistical history. The authorised user (the lecturer or tutor in praxis) has the same possibilities than the anonymous, but furthermore he is able to look at the comments and to browse through single feedback forms. This design decision guarantees, that private comments are not public to all students and students cannot see feedback forms in detail (which providing a higher level of anonymity).

- feedback administration across the net. As for authorised users a user name and password (encrypted) must be supplied. The administrator has the possibility to add, modify or remove lectures. Removing lectures is only possible after storing old statistical data to disk. Furthermore it is possible to administrate users and modify the feedback form. At the moment there is only one kind of feedback form, which means that the modification of the form leads to the modification of all forms of all lectures.

- on-line help available in every mode. It describes the functionality of the system and tells the users how to work with the system (filling in the form, browsing through statistical data or editing the feedback form).

The system itself consists of several Java classes, that are downloaded on demand (see [Fig.2]). At the beginning (browsing an index page that includes the „Start Page Applet") the user has the possibility to choose between filling out a feedback form („Feedback Applet") or to look at results of the feedback („Viewer Applet"). Depending on the user's choice, all necessary classes are downloaded from the http server and the applet tries to connect to the feedback server via sockets. For technical reasons (the standard security manager of Java that is implemented in common Internet browsers allows connections only to the computer, from where the applet comes from) the feedback server must residue on the same computer as the http server.

Every time the user wants to look at different data, a connection to the server is established, the needed data is transferred and then the connection is closed. This approach has two main advantages:

- it is not necessary to transfer the whole data to the client
- every time it is guaranteed that the user receives the latest information.

Every client uses commands for filtering out only that information that is needed for representation on the display. This guarantees short loading times and fast presentation of the data on the display.

4 Design and Implementation of the Feedback Server

Client and feedback server are using sockets for communication. We decided to use a simple (textual based) protocol for transferring commands and data. After a keyword a space is necessary, the hyphen is the semicolon. The server listens on a specified port till a connection is opened. It then starts a new process and resumes listening on the port. The new process now waits for a command, responds to it and then again waits for the next command. After a time-out (default is one minute after the last command) the server kills the process and closes the connection.

This strategy provides following advantages:

- the system is stable because old connections are at least closed after one minute
- the system is easy to debug,
- it is possible to implement textual based clients (GNU Emacs for example),
- the user always gets the latest data.

Some of the commands require prior identification. In this case the client has to send the user name and the encrypted password to the server. The server process then checks the password and changes the process status. User names and passwords can be stored by:

- using the server's database
- using network information services like NIS or NIS+

In addition to storing user names and passwords in the local database the server has the possibility to look up user names and passwords in local or global information systems. This is implemented by a configuration file, that specifies a way in accessing the local or global password database and describes how to filter out the full user name and (encrypted) password.

At the beginning of a new month the server process collects all the data from the last month, generates a statistical representation and stores it in the database. If a client wants to view the statistical history of a lecture

the server looks up in the database if such data is available. If he finds the data, he sends it to the client, otherwise he tells the client, that up to now no data is available.

5 Future Work

Though this version of our client runs without problems, there are a few things that we like to improved in the future:

- multiple feedback sheets,
- more kinds of question types,
- multilingual interface,
- different levels of identification,
- secure communication channel.

At the moment, only one standard form of feedback is supported. That means, that the feedback form is the same for all lectures (even after modifying the form). That is surely enough for universities where there exists one standard form for all lectures. When thinking about a universal use, different forms of feedback sheets should be possible. Analogue to this demand there should be more kinds of question types possible. Entry fields and radio buttons should be extended with check boxes, slide bars, graphical entry fields, or even a voice recorder. Universal use also implies the possibility of using different languages (not only German as in this version).

Another extension is the implementation of different levels of anonymity. As stated above, identification might be a problem for students. Nevertheless, to restrict the number of votes, identification is necessary. A three-level identification will be built into the next release (totally anonymous, registered but anonymous to the lecturer and identified). With identification encryption of the password is not enough. Feedback data is then sensitive data, so that using a secure connection will be necessary.

Even without all the improvements to our system the automatically feedback is surely of big help for students and lecturers. In times, when more and more work is done across the Internet, quality and motivation should always be of main interest - this electronic feedback system should be a brick in the wall of things that should be done to guarantee high quality in education and assessment.

References

[Bollin, Luidolt 95] Andreas Bollin and Christian Luidolt. Automatisches Lehrveranstaltungsfeedback. Technical Report TR-G-95-002-AB, Institute for Software Technology, 1995.

[Bollin 97] Andreas Bollin. Intelligent Hand-in Server and Testing Environment. Technical Report TR-E-97-001-AB, Institute for Software Technology, 1997.

[Kappe 93] Frank Kappe, Hermann Maurer and Nick Sherbakov. Hyper-G - A Universal Hypermedia System. J. EHM (Journal of Educational Multimedia and Hypermedia), 2(1), 1993.

[Lesgold 98] Alan Lesgold. On Evaluating Educational Innovations. Learning Research and Development Center, University of Pittsburgh, March 5, 1998.